

expensive and data in RAM is lost when power is removed. A microprocessor system typically has relatively little ROM and EEPROM, and has 1 to 128 megabytes of RAM, since it is not constrained by what will fit on a single integrated circuit device, and often has access to an external disk memory system that serves as a large writable, non-volatile storage area at a lower cost than EEPROM. However, a microcontroller typically has a small RAM of 0.1 to 2.0 K, 2K to 8K of EEPROM, and 8K – 56K of ROM.--

Please replace the paragraph beginning at page 5, line 3, with the following rewritten paragraph:

--Due to the concern for security, applications written for integrated circuit cards have unique properties. For example, each application typically is identified with a particular owner or identity. Because applications typically are written in a low-level programming language, such as assembly language, the applications are written for a particular type of microcontroller. Due to the nature of low level programming languages, unauthorized applications may access data on the integrated circuit card. Programs written for an integrated circuit card are identified with a particular identity so that if two identities want to perform the same programming function there must be two copies of some portions of the application on the microcontroller of the integrated circuit card.--

Please replace the paragraph beginning at page 6, line 25, with the following rewritten paragraph:

--Among the advantages of the invention are one or more of the following. New applications may be downloaded to a smart card without compromising the security of the smart card. These applications may be provided by different companies loaded at different times using different terminals. Security is not compromised since the applications are protected against unauthorized access of any application code or data by the security features provided by the Java virtual machine. Smart card applications can be created in high level languages such as Java and Eiffel, using powerful mainstream program development tools. New applications can be quickly prototyped and downloaded

A3
canceled

to a smart card in a matter of hours without resorting to soft masks. Embedded systems using microcontrollers can also gain many of these advantages for downloading new applications, high level program development, and rapid prototyping by making use of this invention.--

Please replace the paragraph beginning at page 15, line 24, with the following rewritten paragraph:

A4
10037390-102301

--The terminal 14 can also interact with applications running in the integrated circuit card 10. In some cases, different terminals may be used for these purposes. For example, one kind of terminal may be used to prepare applications, different terminals could be used to download the applications, and yet other terminals could be used to run the various applications. Terminals can be automated teller machines (ATMs), point-of-sale terminals, door security systems, toll payment systems, access control systems, or any other system that communicates with an integrated circuit card or microcontroller.--

Please replace the paragraph beginning at page 18, line 6, with the following rewritten paragraph:

A5
cm.t

--To avoid dynamic linking in the card, all the information that is distributed across several Java class files 24a, 24b, and 24c that form the application 24, are coalesced into one card class file 27 by the process shown in the flowchart in Fig. 5. The first class file to be processed is selected 51a. The constant pool 42 is compacted 51b in the following manner. All objects, classes, fields, methods referenced in a Java class file 24a are identified by using strings in the constant pool 42 of the class file 24a. The card class file converter 26 compacts the constant pool 42 found in the Java class file 24a into an optimized version. This compaction is achieved by mapping all the strings found in the class file constant pool 42 into integers (the size of which is microcontroller architecture dependent). These integers are also referred to as IDs. Each ID uniquely identifies a particular object, class, field or method in the application 20. Therefore, the card class file converter 26 replaces the strings in the Java class file constant pool 42 with